INFORMATION PROCESSOR AND METHOD THEREFOR AND RECORDING MEDIUM

Publication number: JP10301773 (A)

Publication date: 1998-11-13

Inventor(s): TAKEUCHI SHOICHI; NANBA SHINJI + (TAKEUCHI SHOICHI, ; NANBA SHINJI)

Applicant(s): SONY CORP + (SONY CORP)

Classification:

- international: G06F1/00; G06F21/00; G06F21/22; G06F9/54; (IPC1-7): G06F9/06

- European: G06F21/00N3A; G06F21/00N3E1; G06F21/00N7P5M2

Application number: JP19970112182 19970430

Priority number(s): JP19970112182 19970430

Abstract of JP 10301773 (A)

PROBLEM TO BE SOLVED: To execute only a program developed by an authorized software developer in a certain program execution environment. SOLUTION: In a program execution system, an application program enciphered by using a secret key A is supplied to a decoding part 82, and a public key B and a public key A (corresponding to the secret key A) enciphered by using a secret key B corresponding to the public key B are supplied to a decoding part 84. The enciphered public key A is decoded by using the public key B by the decoding part 84, and supplied to the decoding part 82. The enciphered application program is decoded by using the public key A from the decoding part 84 by the decoding part 82, and a Java byte code as the decoded result is supplied to a Java virtual machine 83. The Java byte code from the decoding part 82 is interpreted and executed by a Java virtual machine 83.

(19)日本国特許庁（ＪＰ）　(12) **公開特許公報（Ａ）**　(11)特許出願公開番号

特開平10－301773

(43)公開日　平成10年(1998)11月13日

(51)Int.Cl.[6]	識別記号	ＦＩ	
Ｇ０６Ｆ　9/06	５５０	Ｇ０６Ｆ　9/06	５５０Ａ
			５５０Ｅ
			５５０Ｚ

審査請求　未請求　請求項の数16　ＯＬ　（全 22 頁）

(21)出願番号　特願平9－112182

(22)出願日　平成９年(1997)４月30日

(71)出願人　000002185
ソニー株式会社
東京都品川区北品川６丁目７番35号

(72)発明者　竹内　彰一
東京都品川区北品川６丁目７番35号　ソニー株式会社内

(72)発明者　難波　慎二
東京都品川区北品川６丁目７番35号　ソニー株式会社内

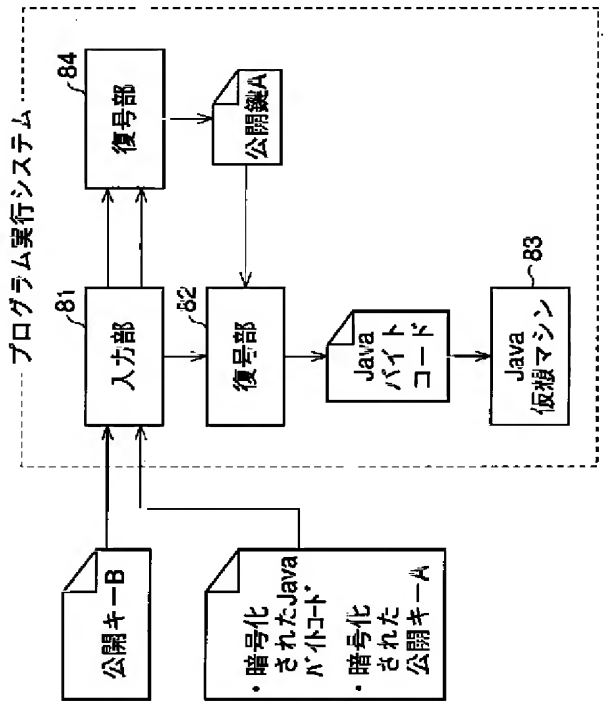
(74)代理人　弁理士　稲本　義雄

(54)【発明の名称】　情報処理装置および情報処理方法、並びに記録媒体

(57)【要約】

【課題】　あるプログラム実行環境において、正当なソフトウェア開発者が開発したプログラムのみが実行されるようにする。

【解決手段】　プログラム実行システムにおいて、秘密キーＡを用いて暗号化されたアプリケーションプログラムは復号部８２に供給され、公開キーＢ、およびその公開キーＢに対応する秘密キーＢを用いて暗号化された公開キーＡ（秘密キーＡに対応するもの）は復号部８４に供給される。復号部８４では、暗号化された公開キーＡが、公開キーＢを用いて復号化され、復号部８２に供給される。復号部８２では、復号部８４からの公開キーＡを用いて、暗号化されたアプリケーションプログラムが復号化され、その復号結果としてのＪａｖａバイトコードが、Ｊａｖａ仮想マシン８３に供給される。Ｊａｖａ仮想マシン８３では、復号部８２からのＪａｖａバイトコードが解釈、実行される。



【特許請求の範囲】
【請求項１】　プログラムを実行するための処理を行う情報処理装置であって、
前記プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化する第１のキー復号化手段と、
前記第１のキー復号化手段により復号化された前記第１のキーを用いて、前記プログラムを暗号化したものを復号化するプログラム復号化手段と、
前記プログラム復号化手段が出力する前記プログラムを実行する実行手段とを備えることを特徴とする情報処理装置。
【請求項２】　前記第２のキーが暗号化されている場合において、
前記第２のキーを暗号化したものを、第３のキーを用いて復号化する第２のキー復号化手段をさらに備えることを特徴とする請求項１に記載の情報処理装置。
【請求項３】　前記第１のキーは、前記プログラムを共通鍵暗号化方式で暗号化するのに用いられた共通キーであり、
前記第２のキーは、前記第１のキーを公開鍵暗号化方式で暗号化するのに用いられた秘密キーに対応する公開キーであり、
前記第３のキーは、前記第２のキーを公開鍵暗号化方式で暗号化するのに用いられた秘密キーに対応する公開キーである。ことを特徴とする請求項２に記載の情報処理装置。
【請求項４】　前記プログラム復号化手段は、複数の前記第１のキーを用いて、前記プログラムを暗号化したものを復号化することを特徴とする請求項１に記載の情報処理装置。
【請求項５】　プログラムを実行するための処理を行う情報処理方法であって、
前記プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化し、
その復号化された前記第１のキーを用いて、前記プログラムを暗号化したものを復号化し、
その復号化された前記プログラムを実行することを特徴とする情報処理方法。
【請求項６】　コンピュータに、
プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化させ、
その復号化された前記第１のキーを用いて、前記プログラムを暗号化したものを復号化させ、
その復号化された前記プログラムを実行させるためのプログラムが記録されていることを特徴とする記録媒体。
【請求項７】　前記第２のキーも記録されていることを特徴とする請求項６に記載の記録媒体。
【請求項８】　プログラムを処理する情報処理装置であって、
請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段を備えることを特徴とする情報処理装置。
【請求項９】　プログラムを処理する情報処理方法であって、
請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化することを特徴とする情報処理方法。
【請求項１０】　請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されていることを特徴とする記録媒体。
【請求項１１】　プログラムを実行するための処理を行う情報処理装置であって、
前記プログラムを実行する実行手段と、
前記プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化するキー復号化手段と、
前記キー復号化手段により復号化された前記第１のキーを用いて、前記プログラムに付されている署名が正当なものかどうかを確認する確認手段と、
前記確認手段により正当なものであることが確認された署名が付されている前記プログラムを、前記実行手段に供給する供給手段とを備えることを特徴とする情報処理装置。
【請求項１２】　プログラムを実行するための処理を行う情報処理方法であって、
前記プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化し、
その復号化された前記第１のキーを用いて、前記プログラムに付されている署名が正当なものかどうかを確認し、
前記署名が正当なものであることが確認された場合のみ、前記プログラムを実行することを特徴とする情報処理方法。
【請求項１３】　コンピュータに、
プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化させ、
その復号化された前記第１のキーを用いて、前記プログラムに付されている署名が正当なものかどうかを確認させ、
前記署名が正当なものであることが確認された場合のみ、前記プログラムを実行させるためのプログラムが記録されていることを特徴とする記録媒体。
【請求項１４】　プログラムを処理する情報処理装置であって、

請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理する処理手段を備えることを特徴とする情報処理装置。
【請求項１５】　プログラムを処理する情報処理方法であって、
請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理することを特徴とする情報処理方法。
【請求項１６】　請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように処理されたプログラムが記録されていることを特徴とする記録媒体。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、情報処理装置および情報処理方法、並びに記録媒体に関し、特に、あるプログラム実行環境において、正当なソフトウェア開発者が開発したプログラムのみが実行されるようにする情報処理装置および情報処理方法、並びに記録媒体に関する。
【０００２】
【従来の技術】最近急速に普及してきたインターネットに適していることから、Ｊａｖａ（米国Sun Microsystems社の商標）が注目されている。Ｊａｖａは、オブジェクト指向言語であるＪａｖａ言語や、そのＪａｖａ言語で記述されたプログラム（以下、適宜、Ｊａｖａプログラムという）の実行に適したプロセッサのアーキテクチャを定義した仮想マシン（以下、適宜、Ｊａｖａ仮想マシンという）、その他のＪａｖａと関連する要素を呼ぶのに、あるいは、それらの総称として用いられる。なお、ここでいう仮想マシンとは、ユーザに対して１台のコンピュータを複数のコンピュータに仮想的に見せる場合の仮想マシンではなく、言語処理系を実装する場合に想定する仮想的なマシンを意味する。
【０００３】Ｊａｖａ仮想マシンは、実際の（現実の）コンピュータ上において、そのコンピュータにインストールされているＯＳ（Operating System）上で動作するように実装される。一方、Ｊａｖａプログラムは、Ｊａｖａ仮想マシンの命令セットからなるバイナリコードにコンパイルされる。このバイナリコードは、Ｊａｖａ仮想マシンが動作するどのようなハードウェアでも実行することができる。従って、Ｊａｖａ仮想マシンさせ動作すれば、コンパイル済みのＪａｖａプログラムは、種々のプラットフォームで実行することができる。
【０００４】
【発明が解決しようとする課題】上述したように、Ｊａｖａ仮想マシンを実装すれば、どのようなマシン上でも、Ｊａｖａプログラムを実行することができること等から、Ｊａｖａ仮想マシンが、多数のユーザに普及することが予想され、さらに、そのような多数のユーザ向けに、多くのアプリケーションプログラムが開発、配布（有償、無償を問わない）されることが予想される。
【０００５】このような状況の下、Ｊａｖａ仮想マシンその他のプログラム実行環境を開発、配布した者からすれば、自身が開発等したプログラム実行環境下において実行される、第三者により開発されたアプリケーションプログラムの配布を制限したい場合がある。即ち、例えば、ライセンス契約を結んだ者だけに、アプリケーションプログラムの配布を許可したい場合がある。
【０００６】一方、Ｊａｖａ仮想マシンでは、Ｊａｖａコンパイラでｊａｖａプログラムを、バイトコード（Ｊａｖａバイトコード）と呼ばれる中間コードにコンパイルしたものが解釈されて実行されるが、Ｊａｖａバイトコードは、それを逆コンパイルすることにより、比較的容易に理解することができるため、リバースエンジニアリングを簡単にすることができる。従って、第三者による模倣や改竄を防止する必要がある。
【０００７】本発明は、このような状況に鑑みてなされたものであり、あるプログラム実行環境下におけるプログラムの実行を制限することができるようにし、さらに、プログラムの模倣や改竄を防止することができるようにもするものである。
【０００８】
【課題を解決するための手段】請求項１に記載の情報処理装置は、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化する第１のキー復号化手段と、第１のキー復号化手段により復号化された第１のキーを用いて、プログラムを暗号化したものを復号化するプログラム復号化手段と、プログラム復号化手段が出力するプログラムを実行する実行手段とを備えることを特徴とする。
【０００９】請求項５に記載の情報処理方法は、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化し、その復号化された第１のキーを用いて、プログラムを暗号化したものを復号化し、その復号化されたプログラムを実行することを特徴とする。
【００１０】請求項６に記載の記録媒体は、コンピュータに、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化させ、その復号化された第１のキーを用いて、プログラムを暗号化したものを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されていることを特徴とする。
【００１１】請求項８に記載の情報処理装置は、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段を備えることを特徴とする。
【００１２】請求項９に記載の情報処理方法は、請求項１に記載の情報処理装置で実行可能なコードに復号化さ

れる暗号文に、プログラムを暗号化することを特徴とする。

【００１３】請求項１０に記載の記録媒体は、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されていることを特徴とする。

【００１４】請求項１１に記載の情報処理装置は、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化するキー復号化手段と、キー復号化手段により復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認する確認手段と、確認手段により正当なものであることが確認された署名が付されているプログラムを、実行手段に供給する供給手段とを備えることを特徴とする。

【００１５】請求項１２に記載の情報処理方法は、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化し、その復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認し、署名が正当なものであることが確認された場合のみ、プログラムを実行することを特徴とする。

【００１６】請求項１３に記載の記録媒体は、コンピュータに、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化させ、その復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認させ、署名が正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されていることを特徴とする。

【００１７】請求項１４に記載の情報処理装置は、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理する処理手段を備えることを特徴とする。

【００１８】請求項１５に記載の情報処理方法は、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理することを特徴とする。

【００１９】請求項１６に記載の記録媒体は、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように処理されたプログラムが記録されていることを特徴とする。

【００２０】請求項１に記載の情報処理装置においては、第１のキー復号化手段は、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化し、プログラム復号化手段は、第１のキー復号化手段により復号化された第１のキーを用いて、プログラムを暗号化したものを復号化するようになされている。実行手段は、プログラム復号化手段が出力するプログラムを実行するようになされている。

【００２１】請求項５に記載の情報処理方法においては、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化し、その復号化された第１のキーを用いて、プログラムを暗号化したものを復号化し、その復号化されたプログラムを実行するようになされている。

【００２２】請求項６に記載の記録媒体には、コンピュータに、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化させ、その復号化された第１のキーを用いて、プログラムを暗号化したものを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されている。

【００２３】請求項８に記載の情報処理装置においては、暗号化手段が、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化するようになされている。

【００２４】請求項９に記載の情報処理方法においては、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化するようになされている。

【００２５】請求項１０に記載の記録媒体には、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されている。

【００２６】請求項１１に記載の情報処理装置においては、キー復号化手段は、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化し、確認手段は、キー復号化手段により復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認するようになされている。供給手段は、確認手段により正当なものであることが確認された署名が付されているプログラムを、実行手段に供給するようになされている。

【００２７】請求項１２に記載の情報処理方法においては、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化し、その復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認し、署名が正当なものであることが確認された場合のみ、プログラムを実行するようになされている。

【００２８】請求項１３に記載の記録媒体には、コンピュータに、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化させ、その復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認させ、署名が正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラ

ムが記録されている。
【００２９】請求項１４に記載の情報処理装置においては、処理手段が、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理するようになされている。
【００３０】請求項１５に記載の情報処理方法においては、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理するようになされている。
【００３１】請求項１６に記載の記録媒体には、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように処理されたプログラムが記録されている。
【００３２】
【発明の実施の形態】以下に、本発明の実施の形態を説明するが、その前に、特許請求の範囲に記載の発明の各手段と以下の実施の形態との対応関係を明らかにするために、各手段の後の括弧内に、対応する実施の形態（但し、一例）を付加して、本発明の特徴を記述すると、次のようになる。
【００３３】即ち、請求項１に記載の情報処理装置は、プログラムを実行するための処理を行う情報処理装置であって、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化する第１のキー復号化手段（例えば、図１４に示す復号部８４や、図２２に示す復号部１３１など）と、第１のキー復号化手段により復号化された第１のキーを用いて、プログラムを暗号化したものを復号化するプログラム復号化手段（例えば、図１４に示す復号部８２や、図２２に示す復号部１３２など）と、プログラム復号化手段が出力するプログラムを実行する実行手段（例えば、図１４や図２２に示すＪａｖａ仮想マシン８３など）とを備えることを特徴とする。
【００３４】請求項２に記載の情報処理装置は、第２のキーが暗号化されている場合において、第２のキーを暗号化したものを、第３のキーを用いて復号化する第２のキー復号化手段（例えば、図２２に示す復号部８４など）をさらに備えることを特徴とする。
【００３５】請求項８に記載の情報処理装置は、プログラムを処理する情報処理装置であって、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段（例えば、図１２に示すプログラムの処理ステップＳ７など）を備えることを特徴とする。
【００３６】請求項１１に記載の情報処理装置は、プログラムを実行するための処理を行う情報処理装置であって、プログラムを実行する実行手段（例えば、図１７に示すＪａｖａ仮想マシン８３など）と、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化するキー復号化手段（例えば、図１７に示す復号部８４など）と、キー復号化手段により復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認する確認手段（例えば、図１７に示す署名確認部１０３など）と、確認手段により正当なものであることが確認された署名が付されているプログラムを、実行手段に供給する供給手段（例えば、図１７に示す仮想マシン入力制御部１０４など）とを備えることを特徴とする。
【００３７】請求項１４に記載の情報処理装置は、プログラムを処理する情報処理装置であって、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理する処理手段（例えば、図１６に示すプログラムの処理ステップＳ２２およびＳ２３など）を備えることを特徴とする。
【００３８】なお、勿論この記載は、各手段を上記したものに限定することを意味するものではない。
【００３９】次に、本発明は、Ｊａｖａのような仮想マシンの他、現実のマシンそのものについても適用可能であるが、ここでは、本発明を、Ｊａｖａ仮想マシンに適用した場合を例に説明する。
【００４０】なお、Ｊａｖａについては、例えば、日経ＢＰ社発行の日経エレクトロニクスの１９９６．３．２５（ｎｏ．６５８）、同１９９６．６．１７（ｎｏ．６６４）などに、その詳細が記載されているので、ここでは、簡単に説明する。
【００４１】Ｊａｖａ仮想マシンは、抽象化された実行機械であり、その実態は、実際の計算機で実行されるプログラムである。そして、Ｊａｖａ仮想マシンは、実際の計算機と同様に、プログラムカウンタや、スタックレジスタ、汎用レジスタ、スタックやヒープとしてのメモリ、その他の資源を有するが、これらの資源は、実際の計算機の資源にマッピングされている。
【００４２】即ち、図１に示すように、いま実際の計算機１が、中央演算処理装置２、その中央演算処理装置２が内蔵するレジスタ３、メモリ４などの資源を有するとして、この計算機１に、Ｊａｖａ仮想マシン１１を実装すると、そのＪａｖａ仮想マシン１１の資源として、実際の計算機１の資源がマッピングされる。図１の実施の形態では、Ｊａｖａ仮想マシン１１は、資源として、レジスタ１３や、メモリ１４などを有しており、レジスタ１３はレジスタ３に、メモリ１４の２００番地は、メモリ４の１００番地に、それぞれマッピングされている。
【００４３】実際の計算機１では、中央演算処理装置２に対する命令は、その資源に対する操作として実行されるが、Ｊａｖａ仮想マシン１１においても、その資源に対する操作として実行される命令が定義されている。このＪａｖａ仮想マシン１１に対する命令を記述するための言語がＪａｖａ言語であり、Ｊａｖａ仮想マシン１１では、このＪａｖａ言語で記述されたソースプログラム

を、Ｊａｖａコンパイラで、Ｊａｖａバイトコードにコンパイルしたものが、解釈、実行される。

【００４４】即ち、図２に示すように、Ｊａｖａ言語で記述されたソースプログラムであるＪａｖａ言語プログラムは、Ｊａｖａコンパイラ２１でコンパイルされ、Ｊａｖａバイトコードとされる。このＪａｖａバイトコードが、Ｊａｖａ仮想マシン１１に入力され、Ｊａｖａ仮想マシン１１では、Ｊａｖａバイトコードが、実際の計算機１（中央演算処理装置２）が解釈することのできる機械語コード（マシン語）に変換される。具体的には、例えば、図３に示すように、「ｍｏｖｅ　＃１２５，レジスタ１３」というＪａｖａバイトコードで記述された「数字の１２５を、レジスタ１３にセット」という命令（Ｊａｖａバイトコード命令）が、Ｊａｖａ仮想マシン１１に入力された場合、Ｊａｖａ仮想マシン１１では、このＪａｖａバイトコードが、「ｍｏｖｅ　＃１２５，レジスタ３」という機械語コードで記述された命令（マシン語命令）に変換される。

【００４５】そして、計算機１では、この機械語コードで記述された命令が実行されることにより、図４（Ａ）に示すように、計算機１のレジスタ３に、数字の１２５がセットされる。

【００４６】上述したように、計算機１のレジスタ３には、Ｊａｖａ仮想マシン１１のレジスタ１３がマッピングされており、従って、計算機１のレジスタ３に、数字の１２５をセットすることは、Ｊａｖａ仮想マシン１１から見れば、図４（Ｂ）に示すように、そのレジスタ１３に数字の１２５をセットすることになる。

【００４７】以上のように、Ｊａｖａ仮想マシン１１へのＪａｖａバイトコードによる命令は、計算機１用の機械語コードに変換され、Ｊａｖａ仮想マシン１１の資源にマッピングされている計算機１の資源に対する操作として実行される。この操作は、Ｊａｖａ仮想マシン１１から見れば、その資源（Ｊａｖａ仮想マシン１１の資源）に対する操作となり、Ｊａｖａバイトコードによる命令が実行されたことになる。

【００４８】従って、前述したように、Ｊａｖａ仮想マシンを、実際の計算機（コンピュータ）に実装することで、その計算機が使用しているＣＰＵ（Central ProcessingUnit）やＯＳに無関係に、コンパイル済みのＪａｖａプログラムを実行することができる。

【００４９】ここで、Ｊａｖａバイトコードを機械語コードに変換して実行する手法としては、例えば、Ｂａｓｉｃ言語によるプログラムを実行する場合に採用されている、各命令を機械語コードに逐次翻訳して実行するインタープリタ形式と、命令を一括して機械語コードに翻訳して実行するＪＩＴ（Just In Time）コンパイラ形式などがある。

【００５０】なお、Ｂａｓｉｃ言語によるプログラムを実行する場合に採用されているインタープリタ形式は、そのソースコードを翻訳する点で、中間コードであるＪａｖａバイトコードを翻訳する場合と異なるが、ここでは、これらを特に区別しない（区別する必要もない）。

【００５１】次に、図５は、本発明を適用したプログラム提供システム（システムとは、複数の装置が論理的に集合したものをいい、各構成の装置が同一筐体中にあるか否かは問わない）の一実施の形態の構成例を示している。

【００５２】このプログラム提供システムにおいては、ソフトウェア開発者が、プログラム認証機関により認証されていないアプリケーションプログラムをユーザに配布した場合に、そのユーザ端末３３上での、そのアプリケーションプログラムの実行を制限するようになされている。

【００５３】即ち、ソフトウェア開発者は、例えば、プログラム認証機関、またはプログラム認証機関にプログラムの認証を依頼している者が開発したプログラム実行環境としてのＪａｖａ仮想マシン上で動作するアプリケーションプログラムを開発し、その配布を希望する場合には、プログラム認証機関との間で、その旨のライセンス契約を結む。ライセンス契約を結ぶと、プログラム認証機関からソフトウェア開発者に対しては、アプリケーションプログラムの認証としての、そのアプリケーションプログラムの暗号化、またはアプリケーションプログラムへの署名の付加に用いるキーが配布される。

【００５４】具体的には、ここでは、アプリケーションプログラムに対する暗号化または署名の付加が、例えば、ＲＳＡ方式（ＭＩＴの３人の研究者により考案されたもので、ＲＳＡは、その３人の研究者の頭文字をとったものである）などに代表される公開鍵暗号化方式によって行われるものとすると、その公開鍵暗号化方式による暗号化に用いられる秘密キーＡと、その秘密キーＡを用いた暗号化結果を復号化するのに用いる公開キーＡとのセットが配布される。

【００５５】ここで、ソフトウェア開発者が有するソフトウェア開発者サーバ３１と、プログラム認証機関が有するプログラム認証機関サーバ３２とは、例えば、インターネットや、公衆回線、ＣＡＴＶ網、地上波、衛星回線、その他でなるネットワーク３４を介して通信することができるようになされており、秘密キーＡおよび公開キーＡのセットの配布は、例えば、このネットワーク３４を介して行われる。

【００５６】なお、秘密キーＡおよび公開キーＡは、プログラム認証機関が用意して、ライセンス契約を結んだソフトウェア開発者に配布する他、ソフトウェア開発者が、自身で用意するようにしても良い。

【００５７】ソフトウェア開発者は、秘密キーＡおよび公開キーＡの配布を受けた後、そのうちの公開キーＡを、例えば、そのソフトウェア開発者サーバ３１から、ネットワーク３４を介して、プログラム認証機関サーバ

３２に送信する。
【００５８】プログラム認証機関サーバ３２は、ソフトウェア開発者サーバ３１から公開キーＡを受信すると、それを暗号化し、その暗号化した公開キー（以下、適宜、暗号化公開キーという）Ａを、ネットワーク３４を介して、ソフトウェア開発者サーバ３１に送信する。
【００５９】ここで、プログラム認証機関サーバ３２において、公開キーＡの暗号化は、例えば、公開鍵暗号化方式により行われる。即ち、プログラム認証機関サーバ３２では、暗号化に用いる秘密キーＢと、その秘密キーＢを用いた暗号化結果を復号化するのに用いる公開キーＢとのセットが用意されており、公開キーＡの暗号化は、その秘密キーＢを用いて行われる。
【００６０】一方、ソフトウェア開発者サーバ３１では、プログラム認証機関サーバ３２からの暗号化公開キーＡが受信され、その後、秘密キーＡを用いての、アプリケーションプログラムの暗号化、またはアプリケーションプログラムへの署名の付加が行われる。そして、暗号化されたアプリケーションプログラム、または署名の付加されたアプリケーションプログラムが、暗号化公開キーＡと対応付けられて記憶される。
【００６１】そして、ソフトウェア開発者サーバ３１は、例えば、ネットワーク３４を介して、ユーザ端末３３からアプリケーションプログラムの要求があると、そのアプリケーションプログラム（上述したように、暗号化されたアプリケーションプログラム、または署名の付加されたアプリケーションプログラム）を、それに対応付けられた暗号化公開キーＡとともに、ネットワーク３４を介して、ユーザ端末３３に送信する。
【００６２】ユーザ端末３３には、プログラム認証機関、またはプログラム認証機関にプログラムの認証を依頼している者が開発したプログラム実行環境としてのＪａｖａ仮想マシンが実装されている。即ち、プログラム認証機関サーバ３２には、そのプログラム認証機関が提供するＪａｖａ仮想マシンを含む、Ｊａｖａプログラムの実行環境としてのプログラム実行システム（コンピュータ（ユーザ端末３３）を、プログラム実行システムとして動作させるためのプログラム）が記憶されており、ユーザが、ユーザ端末３３を操作することにより、そのプログラム実行システムを、プログラム認証機関サーバ３２に要求すると、プログラム認証機関サーバ３２は、プログラム実行システムを、例えば、ネットワーク３４を介して、ユーザ端末３３に送信する。ユーザ端末３３においては、このようにしてプログラム認証機関から提供されるプログラム実行システムが実装されている。
【００６３】なお、プログラム認証機関サーバ３２からユーザ端末３３に対しては、プログラム実行システムとともに、公開キーＢも送信されるようになされており、この公開キーＢは、ユーザ端末３３において記憶されるようになされている。
【００６４】そして、プログラム実行システムが実装された、Ｊａｖａ仮想マシンとしてのユーザ端末３３では、ソフトウェア開発者サーバ３１から受信したアプリケーションプログラムが、プログラム認証機関によって認証されているものであるときのみ、そのアプリケーションプログラムが、プログラム実行システム上において、正常に実行される。
【００６５】即ち、ユーザ端末３３では、ソフトウェア開発者サーバ３１から受信した暗号化公開キーＡが、プログラム認証機関サーバ３２から受信した公開キーＢを用いて、公開キーＡに復号化される。さらに、ユーザ端末３３では、その公開キーＡを用いて、ソフトウェア開発者サーバ３１から受信した、暗号化されたアプリケーションプログラムが復号化され、その復号化されたアプリケーションプログラムが、Ｊａｖａ仮想マシン上で実行される。あるいは、また、ソフトウェア開発者サーバ３１から受信したアプリケーションプログラムに付加された署名の正当性が、公開キーＡを用いて確認され、署名の正当性が確認された場合のみ、アプリケーションプログラムが、Ｊａｖａ仮想マシン上で実行される。
【００６６】従って、公開キーＢを用いて復号化されるキーが、プログラム認証機関から発行された、いわば正当な暗号化公開キーＡではない場合、その復号化結果として、公開キーＡを得ることはできない。このため、そのような復号化結果を用いて、アプリケーションプログラムを復号化しても、Ｊａｖａ仮想マシン上で正常な処理が行われるようなアプリケーションプログラムを得られず、結局、Ｊａｖａ仮想マシンは正常動作しない。
【００６７】同様に、公開キーＡでないキーを用いて署名を確認した場合には、その署名の正当性は否定されるため、やはり、Ｊａｖａ仮想マシンでは、アプリケーションプログラムは実行されない。
【００６８】以上のように、ソフトウェア開発者サーバ３１から受信したアプリケーションプログラムが、プログラム認証機関によって認証されていない場合には、Ｊａｖａ仮想マシンとしてのユーザ端末３３では、そのアプリケーションプログラムは実行されない（実行されても、正常には実行されない）。
【００６９】従って、結果として、プログラム実行環境としてのＪａｖａ仮想マシンを開発、配布した者は、そのＪａｖａ仮想マシンにおいて実行される、第三者により開発されたアプリケーションプログラムの、いわば勝手な配布を制限し、例えば、ライセンス契約を結んだソフトウェア開発者だけに、アプリケーションプログラムの配布を許可することができる。
【００７０】さらに、上述の場合においては、ソフトウェア開発者は、公開キーＡおよび秘密キーＡ、並びに暗号化公開キーＡを得た後は、ユーザ端末３３に実装されたＪａｖａ仮想マシンで実行可能なアプリケーションプログラムの配布を自由に行うことができる。従って、新

たなアプリケーションプログラムを開発するごとに、そのアプリケーションプログラムを、プログラム認証機関サーバ３２に送信して認証してもらう手間を省くことができる。

【００７１】但し、プログラム認証機関側からすれば、ソフトウェア開発者による公開キーＡおよび秘密キーＡ、並びに暗号化公開キーＡの使用を、幾つかのアプリケーションプログラムに制限したい場合があるが、これは、例えば、プログラム認証機関が、公開キーＡを暗号化するのに用いる秘密キーＢを変更することによって対処可能である（この場合、ユーザには、その変更後の秘密キーに対する公開キーを配布する必要がある）。また、そのような変更を行わないまでも、公開キーＡおよび秘密キーＡとして、ソフトウェア開発者ごとにユニークなものを割り当てるようにすれば、暗号化公開キーＡや、署名を参照することにより、アプリケーションプログラムを配布したソフトウェア開発者を特定することができる。従って、例えば、公開キーＡおよび秘密キーＡ、並びに暗号化公開キーＡの使用を、幾つかのアプリケーションプログラムに制限する旨のライセンス契約をしたのにも拘らず、その数を越えたアプリケーションプログラムの配布を行っているソフトウェア開発者を、容易に特定することができる。

【００７２】なお、ソフトウェア開発者は、アプリケーションプログラムを、例えば、ＣＤ（Compact Disc）－ＲＯＭ、磁気ディスク、その他でなる記録媒体３５に記録して、郵送、店頭における販売、その他の手段で、ユーザに配布することも可能であるが、この場合においても、上述の場合と同様に、そのアプリケーションプログラムが、プログラム認証機関によって認証されていないときには、Ｊａｖａ仮想マシンとしてのユーザ端末３３では、そのアプリケーションプログラムは実行されない。

【００７３】また、上述の場合においては、ソフトウェア開発者とプログラム認証機関との間で、ネットワーク３４を介して、データ（ここでは、公開キーＡおよび秘密キーＡ、並びに暗号化公開キーＡ）のやりとりを行うようにしたが、その他、データのやりとりは、そのデータを記録した記録媒体（図示せず）を郵送などすることによって行うことも可能である。

【００７４】さらに、プログラム認証機関からユーザへのプログラム実行システムおよび公開キーＢの提供も、例えば、ＣＤ－ＲＯＭ、磁気ディスク、その他でなる記録媒体３６に記録して、郵送、店頭における販売、その他の手段で行うことが可能である。

【００７５】さらに、図５の実施の形態では、ソフトウェア開発者サーバ３１、プログラム認証機関サーバ３２、またはユーザ端末３３を、それぞれ１つしか設けていないが、これらは、複数設けることが可能である。

【００７６】次に、図６は、図５のソフトウェア開発者サーバ３１の構成例を示している。

【００７７】ＣＰＵ４１は、補助記憶装置４６に記憶（記録）されたオペレーティングシステムの制御の下、同じく補助記憶装置４６に記憶されたプログラムを実行することで、各種の処理を行う。ＲＯＭ（Read Only Memory）４２は、例えば、ＩＰＬ（Initial Program Loading）のプログラムなどを記憶している。ＲＡＭ（Random Access Memory）４３は、ＣＰＵ４１が実行するプログラムや、ＣＰＵ１の動作上必要なデータを記憶する。入力部４４は、例えば、キーボードやマウスなどで構成され、所定のデータやコマンドを入力するときなどに操作される。出力部４５は、ディスプレイやプリンタなどで構成され、所定の情報を表示、印刷する。補助記憶装置４６は、例えば、ハードディスクなどで構成され、オペレーティングシステム、その他のＣＰＵ４１が実行するプログラムなどを記憶している。さらに、補助記憶装置４６は、ＣＰＵ４１の処理結果その他の必要なデータなども記憶する。通信制御部４７は、ネットワーク３４を介して行われる通信を制御する。

【００７８】次に、図７は、図５のプログラム認証機関サーバ３２の構成例を、図８は、図５のユーザ端末３３の構成例を、それぞれ示している。

【００７９】プログラム認証機関サーバ３２は、ＣＰＵ５１乃至通信制御部５７で構成され、また、ユーザ端末３３は、ＣＰＵ６１乃至通信制御部６７で構成され、これらは、図６におけるＣＰＵ４１乃至通信制御部４７とそれぞれ同様に構成されるので、その説明は省略する。

【００８０】次に、公開キーＡやアプリケーションプログラムの暗号化／復号化の方法の１つとしての、上述した公開鍵暗号化方式による暗号化／復号化について、さらに説明する。

【００８１】図９は、公開鍵暗号化方式による暗号化／復号化システムの構成例を示している。

【００８２】暗号化器７１には、暗号化対象である平文が入力される。そして、暗号化器７１では、秘密キーと呼ばれる各個人に特有の暗号キーを用いて、平文が暗号化され、暗号文とされる。

【００８３】一方、復号化器７２には、暗号化器７１で暗号化された暗号文が入力される。そして、復号化器７２では、公開キーと呼ばれる広く一般に公開される復号キーを用いて、暗号文が復号化され、元の平文とされる。

【００８４】ソフトウェア開発者サーバ３１では、アプリケーションプログラムとしてのＪａｖａバイトコードが、暗号化器７１における場合と同様にして、秘密キーＡを用いて暗号化される。また、プログラム認証機関サーバ３２では、ソフトウェア開発者サーバ３１からの公開キーＡが、やはり暗号化器７１における場合と同様にして、秘密キーＢを用いて暗号化される。

【００８５】一方、ユーザ端末３３では、復号化器７２

における場合と同様にして、暗号化公開キーＡが、公開キーＢを用いて復号化され、さらに、その復号化された公開キーＡを用いて、暗号化されたアプリケーションプログラムが復号化される。

【００８６】ここで、暗号化／復号化の手法は、公開鍵暗号化方式に限定されるものではなく、その他、例えば、ＤＥＳ（Data Encryption Standard）方式（ＩＢＭ社により開発され、米国連邦政府の標準として実用化されたもの）などに代表される共通鍵暗号化方式その他の方式を採用することが可能である。なお、共通鍵暗号化方式では、当事者間以外に公開されない共通キーを用いて暗号化／復号化が行われる。公開鍵暗号化方式では、暗号化に用いられる秘密キーと復号化に用いられる公開キーとは異なるが（逆に、公開キーを暗号化に、秘密キーを復号化に、それぞれ用いてもかまわない）、共通鍵暗号化方式では、暗号化と復号化に、同一のキーである共通キーが用いられるので、その共通キーは、当事者間以外の者には秘密にしておく必要がある。

【００８７】次に、アプリケーションプログラムに署名（ディジタルサイン）を付加する方法の１つとしての公開鍵暗号化方式を利用した署名付加方法について説明する。

【００８８】図１０は、公開鍵暗号化方式を利用した署名の作成／確認を行う暗号化／復号化システムの構成例を示している。

【００８９】ダイジェスト作成器９１には、暗号化対象である平文が入力される。そして、ダイジェスト作成器９１では、入力された平文のダイジェストが、例えば、ＭＤ５やＳＨＡ－１などのアルゴリズムにしたがって作成される。

【００９０】ここで、ダイジェストは、平文の機械的な凝縮文に相当し、入力としての平文が異なれば、そのダイジェストも異なるものが作成される。ダイジェストの作成は、平文を、例えば、ハッシュ関数を用いて変換することで行われる。

【００９１】なお、データベースの検索を行うのに用いるキーワードがとり得る範囲の集合を、ある限られた数値範囲（レコード番号や配列の添字などに対応する）に写像する方法は、ハッシング（hashing）と呼ばれるが、この写像を行う変換関数がハッシュ関数である。

【００９２】ダイジェスト作成器９１で作成されたダイジェストは、暗号化器９２に供給される。暗号化器９２では、例えば、図９の暗号化器７１における場合と同様にして、ダイジェストが、秘密キーを用いて暗号化され、この暗号化されたダイジェストがディジタルサイン（署名）として出力される。そして、ディジタルサインが、元の平文に付加され、署名付き平文として出力される。

【００９３】一方、復号化器９３には、署名付き平文を構成するディジタルサインが、また、ダイジェスト作成器９４には、その残りの平文が、それぞれ入力される。復号化器９３では、例えば、図９の復号化器７２における場合と同様にして、ディジタルサインが、公開キーを用いて復号化され、ダイジェストとされる。このダイジェストは、署名確認器９５に供給される。

【００９４】ダイジェスト作成器９４では、ダイジェスト作成器９１における場合と同様にして、そこに入力される平文のダイジェストが作成され、署名確認器９５に供給される。

【００９５】署名確認器９５では、署名（ディジタルサイン）の正当性が判断される（署名の確認が行われる）。即ち、署名確認器９５では、復号化器９３が出力するダイジェストが、ダイジェスト作成器９４が出力するダイジェストと一致するかどうかが確認される。復号化器９３が出力するダイジェストが、ダイジェスト作成器９４が出力するダイジェストと一致しない場合、例えば、平文の改竄が行われたとして、あるいは、復号化器９３で用いられた公開キーが正しくないとして、署名の正当性が否定される。

【００９６】一方、復号化器９３が出力するダイジェストが、ダイジェスト作成器９４が出力するダイジェストと一致する場合、平文の改竄が行われておらず、かつ復号化器９３で用いられた公開キーが正しいとして、署名の正当性が確認される。

【００９７】署名確認器９５には、署名付き平文を構成する平文も供給されるようになされており、そこでは、署名の正当性が確認されると、その平文が出力される。

【００９８】ソフトウェア開発者サーバ３２において、アプリケーションプログラムに、その署名が付される場合においては、上述したように、アプリケーションプログラムのダイジェストが作成され、そのダイジェストが秘密キーＡを用いて暗号化されることにより、署名（ディジタルサイン）が作成される。一方、ユーザ端末３３においては、その署名が公開キーＡを用いてダイジェストに復号化されるとともに、アプリケーションプログラムからダイジェストが作成され、その２つのダイジェストが一致するかどうかで、署名の正当性が確認される。

【００９９】署名は、その署名をした者（ここでは、ソフトウェア開発者）を証明するものであるから、署名が付加されたアプリケーションプログラムについては、その署名から、その出所を、容易に特定することができる。従って、例えば、アプリケーションプログラムにバグなどがある場合に、そのようなバグのあるアプリケーションプログラムを配布したソフトウェア開発者を、容易に見つけ出すことができる。

【０１００】さらに、署名が付加されたアプリケーションプログラムが改竄されていたり、いわゆるコンピュータウイルスなどに侵されている場合においては、そのアプリケーションプログラムから作成されるダイジェストと、署名を復号化して得られるダイジェストとは一致せ

ず、署名の正当性が否定されるから、そのような改竄やコンピュータウイルスに侵されたアプリケーションプログラムが、ユーザ端末３３において実行されることを防止することが可能となる。

【０１０１】また、署名の作成に用いられる秘密キーＡは、第三者（ここでは、ソフトウェア開発者およびプログラム認証機関以外の者）に秘密にされるものであるから、仮に、第三者が、アプリケーションプログラムの解読などを行っても、署名の作成方法まで知ることは困難である。

【０１０２】なお、署名の作成／確認方法は、上述した公開鍵暗号化方式を利用したものに限定されるものではない。

【０１０３】次に、アプリケーションプログラムの認証が、秘密キーＡを用いた暗号化により行われる場合のソフトウェア開発者サーバ３１、プログラム認証機関サーバ３２、およびユーザ端末３３の処理について説明する。

【０１０４】まず、図１１および図１２のフローチャートを参照して、ソフトウェア開発者サーバ３１の処理について説明する。

【０１０５】ソフトウェア開発者は、プログラム認証機関とライセンス契約を結び、公開キーＡおよび秘密キーＡを、例えば、プログラム認証機関から発行してもらう等して取得する。そして、ソフトウェア開発者サーバ３１では、まず最初に、図１１のステップＳ１において、通信制御部４７が、公開キーＡをプログラム認証機関に認証してもらうために、ネットワーク３４を介して、プログラム認証機関サーバ３２に送信し、ステップＳ２に進む。ステップＳ２では、ＣＰＵ４１において、認証された公開キーＡとしての、その公開キーＡを暗号化した暗号化公開キーＡが、プログラム認証機関サーバ３２から送信されてきたかどうかが判定され、まだ、送信されてきていないと判定された場合、ステップＳ２に戻る。

【０１０６】また、ステップＳ２において、暗号化公開キーＡが送信されてきたと判定された場合、ステップＳ３に進み、通信制御部４７において、その暗号化公開キーＡが受信され、ステップＳ４に進む。ステップＳ４では、通信制御部４７で受信された暗号化公開キーが、補助記憶装置４６に転送されて記憶され、処理を終了する。

【０１０７】その後、ソフトウェア開発者が、Ｊａｖａ仮想マシン上で実行されるアプリケーションプログラムを開発し、そのアプリケーションプログラムが、例えば、補助記憶装置４６に記録（記憶）されると、ソフトウェア開発者サーバ３１では、図１２のステップＳ６において、ＣＰＵ４１が、補助記憶装置４６に記憶されたアプリケーションプログラムを、Ｊａｖａコンパイラのプログラムにしたがってコンパイルし、Ｊａｖａバイトコードとする。このＪａｖａバイトコードは、再び、補助記憶装置４６に供給されて記憶される。

【０１０８】そして、ステップＳ７に進み、ＣＰＵ４１において、ステップＳ６のコンパイルの結果得られたＪａｖａバイトコードが、例えば、図９で説明したように、秘密キーＡを用いて暗号化され、暗号化バイトコードとされる。この暗号化バイトコードは、ステップＳ８において、暗号化公開キーＡと対応付けられて、補助記憶装置４６に記憶され、処理を終了する。

【０１０９】次に、図１３のフローチャートを参照して、プログラム認証機関サーバ３２の処理について説明する。

【０１１０】プログラム認証機関は、例えば、プログラム実行環境としてのＪａｖａ仮想マシンを開発した者、あるいは、その依頼を受けた者の機関で、そのプログラム認証機関サーバ３２では、例えば、ライセンス契約を結んだ者のアプリケーションプログラムの認証として、公開キーＡの認証を行う認証処理を行うようになされている。

【０１１１】即ち、プログラム認証機関サーバ３２のＣＰＵ５１では、まず最初に、ステップＳ１１において、例えば、ソフトウェア開発者サーバ３１などから、認証対象の公開キーＡが、ネットワーク３４を介して送信されてきたかどうかが判定され、送信されてきていないと判定された場合、ステップＳ１１に戻る。また、ステップＳ１１において、公開キーＡが送信されてきたと判定された場合、ステップＳ１２に進み、その公開キーＡが、例えば、ライセンス契約を結んだ、いわば正規のソフトウェア開発者からのものであるかどうかが、ＣＰＵ５１によって判定される。

【０１１２】即ち、プログラム認証機関は、Ｊａｖａ仮想マシン上で実行されるアプリケーションプログラムの開発、配布等を許可することについてのライセンス契約を、ソフトウェア開発者との間で結ぶと、そのソフトウェア開発者に対して、例えば、ＩＤおよびパスワードを発行するようになされている。そして、このようなライセンス契約を結んだ正規のソフトウェア開発者からは、認証対象の公開キーＡとともに、ライセンス契約時に発行されたＩＤおよびパスワードが送信されるようになされており、ステップＳ１２では、このＩＤおよびパスワードに基づいて、公開キーＡが、正規のソフトウェア開発者からのものであるかどうかが判定される。

【０１１３】ステップＳ１２において、公開キーＡが、正規のソフトウェア開発者からのものでないと判定された場合、即ち、ライセンス契約を結んでいないソフトウェア開発者から公開キーＡが送信されてきた場合、ステップＳ１３に進み、通信制御部５７において、そのソフトウェア開発者に対して、ライセンス契約を結ばなければ、公開キーＡを認証することができない旨のメッセージが送信され、処理を終了する。

【０１１４】一方、ステップＳ１２において、公開キー

Ａが、正規のソフトウェア開発者からのものであると判定された場合、ステップＳ１４に進み、ＣＰＵ５１において、その公開キーＡが暗号化され、これにより暗号化公開キーＡとされることで、その認証が行われる。

【０１１５】そして、ステップＳ１５に進み、通信制御部５７において、公開キーＡの認証結果としての暗号化公開キーＡが、その公開キーＡを送信してきたソフトウェア開発者、即ち、ここでは、例えば、ソフトウェア開発者サーバ３１に、ネットワーク３４を介して送信され、処理を終了する。

【０１１６】次に、図１４は、ユーザ端末３３においてアプリケーションプログラムを実行するプログラム実行環境としてのプログラム実行システムの機能的構成例を示している。

【０１１７】入力部８１は、暗号化バイトコード（暗号化されたＪａｖａバイトコード）および暗号化公開キーＡ（暗号化された公開キーＡ）、並びに公開キーＢを受け付け、暗号化バイトコードを復号部８２に、暗号化公開キーＡおよび公開キーＢを復号部８４に、それぞれ供給するようになされている。復号部８２は、図１１の復号化器７２としての処理を行うもので、入力部８１からの暗号化バイトコードを、復号部８４から出力される公開キーＡを用いて復号化し、元のＪａｖａバイトコードとするようになされている。復号部８２で得られたＪａｖａバイトコードは、Ｊａｖａ仮想マシン８３に供給されるようになされており、Ｊａｖａ仮想マシン８３は、復号部８２からのＪａｖａバイトコードにしたがった処理を実行するようになされている。復号部８４も、復号部８２と同様に、図１１の復号化器７２としての処理を行うもので、入力部８１からの暗号化公開キーＡを、同じく入力部８１からの公開キーＢを用いて、公開キーＡに復号化し、復号部８２に供給するようになされている。

【０１１８】以上のように構成されるプログラム実行システムでは、まず最初に、入力部８１において、暗号化バイトコードおよび暗号化公開キーＡ、並びに公開キーＢが取得される。即ち、例えば、あらかじめ、ソフトウェア開発者サーバ３１から、ネットワーク３４を介して、暗号化バイトコードおよび暗号化公開キーＡを受信し、それらが、ファイルとして、補助記憶装置６７に記憶されている場合や、暗号化バイトコードおよび暗号化公開キーＡがファイルとして記録された記録媒体３５がユーザ端末３３にセットされる場合などには、入力部８１は、そのファイルをオープンし、暗号化バイトコードおよび暗号化公開キーＡを読み出す。

【０１１９】また、例えば、ソフトウェア開発者サーバ３１が、インターネットとしてのネットワーク３４に接続されており、そのようなソフトウェア開発者サーバ３１において、暗号化バイトコードおよび暗号化公開キーＡが、ＵＲＬ（Uniform Resource Locator）と対応付けられている場合に、ユーザが、入力部６４を操作して、そのＵＲＬを指定したときには、入力部８１は、ソフトウェア開発者サーバ３１からネットワーク３４を介して送信されてくる暗号化バイトコードおよび暗号化公開キーＡを受信する。

【０１２０】さらに、例えば、ソフトウェア開発者サーバ３１が、地上波や衛星回線としてのネットワーク３４を介して、暗号化バイトコードおよび暗号化公開キーＡをディジタル放送などしている場合には、入力部８１は、その放送されてくる暗号化バイトコードおよび暗号化公開キーＡを受信する。

【０１２１】入力部８１は、同様にして、プログラム認証機関が発行する公開キーＢも取得する。

【０１２２】入力部８１は、以上のようにして取得した暗号化バイトコードおよび暗号化公開キーＡ、並びに公開キーＢのうち、暗号化バイトコードを復号部８２に、暗号化公開キーＡおよび公開キーＢを復号部８４に、それぞれ供給する。

【０１２３】復号部８４では、入力部８１からの暗号化公開キーＡが、同じく入力部８１からの公開キーＢを用いて、公開キーＡに復号化され、復号部８２に供給される。復号部８２では、復号部８４からの公開キーＡを用いて、入力部８１からの暗号化バイトコードが復号化され、その復号結果としてのＪａｖａバイトコードが、Ｊａｖａ仮想マシン８３に供給される。Ｊａｖａ仮想マシン８３では、復号部８２からのＪａｖａバイトコードが解釈、実行される。

【０１２４】以上のように、復号部８４において、プログラム認証機関サーバ３２での暗号化に用いられた秘密キーＢに対応する公開キーＢ（秘密キーＢと対になる公開キーＢ）を用いて、暗号化公開キーＡが、公開キーＡに復号化される。さらに、復号部８２において、ソフトウェア開発者サーバ３１での暗号化に用いられた秘密キーＡに対応する公開キーである、復号部８４で復号化された公開キーＡを用いて、暗号化バイトコードの復号化が行われ、その復号結果としてのＪａｖａバイトコードが、Ｊａｖａ仮想マシン８３に入力される。

【０１２５】従って、公開キーＢを用いて復号化されるキーが、プログラム認証機関から発行された、いわば正当な暗号化公開キーＡではない場合、即ち、例えば、暗号化されていない公開キーＡや、プログラム認証機関サーバ３２における暗号化アルゴリズムと異なるアルゴリズムで暗号化された暗号化キーＡ、あるいはそれと同一のアルゴリズムで暗号化されていたとしても、本来使用すべき秘密キーＢを用いずに暗号化された公開キーＡなどが入力された場合には、その復号化結果として、正当な公開キーＡを得ることはできない（正当な公開キーＡが、復号部８４から偶然に出力されることは有り得ないことではないが、ほとんどないに等しい）。このため、そのような復号化結果を用いて、暗号化バイトコードを

復号化しても、Ｊａｖａ仮想マシン８３上で正常な処理が行われるようなアプリケーションプログラムを得られず、結局、Ｊａｖａ仮想マシン８３は正常動作しない。よって、結果として、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードであって、プログラム認証機関で認証されていないものの、Ｊａｖａ仮想マシン８３が実装されたユーザ端末３３を有するユーザへの配布を制限することができる。

【０１２６】即ち、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードの、そのＪａｖａ仮想マシン８３が実装されたユーザ端末３３を有するユーザへの配布は、プログラム認証機関とライセンス契約を結んだソフトウェア開発者にのみ許可することができ、Ｊａｖａ仮想マシン８３の開発者等は、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードの配布を希望するソフトウェア開発者から、いわば、Ｊａｖａ仮想マシン８３を利用するためのライセンス料を得ることが可能となる。

【０１２７】なお、Ｊａｖａ仮想マシン８３に対するＪａｖａバイトコードの入力は、復号部８２からのみ行うことができるようにしておく必要があり、また、復号部８２への公開キーＡの入力も復号部８４からのみ行うことができるようにしておくのが望ましい。

【０１２８】ここで、図１４における復号部８２および８４では、何らかの入力があると、その入力に対して復号化処理が施され、その処理結果が出力される。従って、公開キーＢでないものや、秘密キーＡで暗号化されていないＪａｖａバイトコード、秘密キーＢで暗号化されていない公開キーＡが入力され、そのようなものから得られる復号化結果が、Ｊａｖａ仮想マシン８３に供給されると、Ｊａｖａ仮想マシン８３は、通常暴走する。そこで、復号部８２の出力が、正当（正常）なＪａｖａバイトコードかどうかを確認し、正当なＪａｖａバイトコードである場合にのみ、そのＪａｖａバイトコードを、Ｊａｖａ仮想マシン８３に解釈、実行させるようにすることが可能である。即ち、例えば、Ｊａｖａバイトコードには、その先頭に、マジック（ｍａｇｉｃ）と呼ばれる３２ビットのデータが配置されており、これが本来の値（１６進数で、ＣＡＦＥＢＡＢＥ）である場合に、復号部８２の出力が正当なＪａｖａバイトコードであるとして、Ｊａｖａ仮想マシン８３に解釈、実行させるようにすることができる。この場合、Ｊａｖａ仮想マシン８３の暴走を防止することができる。

【０１２９】ところで、図１４の復号部８２および８４における復号化アルゴリズムや、公開キーＡ，Ｂが、ライセンス契約をしている者以外の者に知られても、Ｊａｖａ仮想マシン８３上におけるアプリケーションプログラムの実行の制限の観点からは、暗号化アルゴリズムやその暗号化に用いる秘密キーＡ，Ｂさえ知られなければ、特に問題はない。即ち、暗号文の復号化方法が知られても、正常に実行可能なＪａｖａバイトコードをＪａｖａ仮想マシン８３に供給するために復号部８２に与えるべき暗号化バイトコードの作成方法や、復号部８４に与えるべき暗号化公開キーＡの作成方法を知られなければ、仮想マシン８３上におけるアプリケーションプログラムの実行を制限することができる。

【０１３０】しかしながら、暗号文の復号化方法が知られた場合、暗号化バイトコードから、元のＪａｖａバイトコードを得ることができる。Ｊａｖａバイトコードは、前述したように、それを逆コンパイルすることにより、その内容を比較的容易に理解することができるため、リバースエンジニアリングを簡単にすることができる。

【０１３１】そこで、このようなリバースエンジニアリングを防止すべく、暗号文の復号化方法は秘密にすることができる。即ち、例えば、暗号化バイトコードの復号化に用いる公開キーＡを暗号化した暗号化公開キーＡを復号化するための公開キーＢは、一般に公開されるものであるが、これを秘密にすることができる。

【０１３２】図１５は、公開キーＢを秘密にするようにしたプログラム実行システムの構成例を示している。なお、図中、図１４における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【０１３３】この実施の形態においては、例えば、Ｊａｖａ仮想マシン８３を含むプログラム実行システムを構成するプログラムの中の１カ所に、または複数箇所に分散して、公開キーＢが配置されており、復号部８４では、その公開キーＢを用いて、暗号文の復号化が行われる。従って、この場合、公開キーＢは、プログラム実行システムの外部に漏れることがなく、その結果、暗号文が、不正に復号され、リバースエンジニアリングされることを防止すること（リバースエンジニアリングされる確率を低減すること）が可能となる。

【０１３４】次に、アプリケーションプログラムの認証が、秘密キーＡを用いて作成された署名が付加されることにより行われる場合のソフトウェア開発者サーバ３１およびユーザ端末３３の処理について説明する。なお、この場合、プログラム認証機関サーバ３２では、上述した、アプリケーションプログラムが暗号化される場合と同様の処理が行われるので、その説明は省略する。

【０１３５】まず、図１６のフローチャートを参照して、ソフトウェア開発者サーバ３１の処理について説明する。なお、ソフトウェア開発者サーバ３１では、既に、図１１のフローチャートで説明した処理が行われ、これにより、プログラム認証機関から暗号化公開キーＡを取得しているものとする。また、ソフトウェア開発者においては、Ｊａｖａ仮想マシン上で実行されるアプリケーションプログラムが開発され、補助記憶装置４６に記憶されているものとする。

【０１３６】この場合も、ソフトウェア開発者サーバ３１では、図１２のステップＳ６における場合と同様に、ステップＳ２１において、ＣＰＵ４１が、補助記憶装置４６に記憶されたアプリケーションプログラムを、Ｊａｖａコンパイラのプログラムにしたがってコンパイルし、Ｊａｖａバイトコードとする。このＪａｖａバイトコードは、再び、補助記憶装置４６に供給されて記憶される。

【０１３７】そして、ステップＳ２２以下に順次進み、ステップＳ２１のコンパイルの結果得られたＪａｖａバイトコードに対して、それが正当なものであることを証明する署名（ディジタルサイン）が付される。

【０１３８】即ち、ステップＳ２２では、ＣＰＵ５１において、Ｊａｖａバイトコードのダイジェストが、例えば、図１０のダイジェスト作成器９１における場合と同様にして作成され、ステップＳ２３に進む。ステップＳ２３では、ＣＰＵ５１において、ステップＳ２２で作成されたダイジェストが、例えば、図１０の暗号化器９２における場合と同様にして、秘密キーＡを用いて暗号化されることにより、ディジタルサイン（署名）が作成される。そして、ステップＳ２４に進み、そのディジタルサインが、Ｊａｖａバイトコードに付される（このようにディジタルサイン（署名）が付されたＪａｖａバイトコードを、以下、適宜、署名付きバイトコードという）。さらに、ステップＳ２４では、署名付きバイトコードが、暗号化公開キーＡと対応付けられて、補助記憶装置４６に記憶され、処理を終了する。

【０１３９】次に、図１７は、ユーザ端末３３においてアプリケーションプログラムの正当性を確認し、正当なもののみを実行するプログラム実行環境としてのプログラム実行システムの機能的構成例を示している。なお、図中、図１４おける場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【０１４０】入力部１０１は、基本的には、図１２における入力部８１と同様に、そこへの入力を受け付けるようになされている。但し、入力部１０１には、暗号化バイトコードの代わりに、署名付きバイトコード（署名（ディジタルサイン）が付されたＪａｖａバイトコード）が入力されるようになされており、そこでは、署名とＪａｖａバイトコードとに分離されて出力されるようになされている。署名は、署名確認部１０３に、Ｊａｖａバイトコードは、メッセージダイジェストシステム１０２および仮想マシン入力制御部１０４に、それぞれ供給されるようになされている。

【０１４１】メッセージダイジェストシステム１０２は、図１０のダイジェスト作成器９４と同様の処理を行うもので、入力部１０１からのＪａｖａバイトコードからダイジェストを作成し、署名確認部１０３に供給するようになされている。署名確認部１０３は、図１０の復号化器９３および署名確認器９５に相当するもので、入力部１０１からの署名の正当性を確認するようになされている。

【０１４２】即ち、署名確認部１０３には、入力部１０１から署名が、メッセージダイジェストシステム１０２からダイジェストが、それぞれ供給される他、復号部８４から、公開キーＡが供給されるようになされている。そして、署名確認部１０３は、その公開キーＡを用いて、署名を復号化することにより、ダイジェストとし、そのダイジェストと、メッセージダイジェストシステム１０２からのダイジェストとを比較することで、署名の正当性を確認するようになされている。さらに、署名確認部１０３は、その確認結果に対応して、仮想マシン入力制御部１０４を制御するようになされている。

【０１４３】仮想マシン入力制御部１０４は、署名確認部１０３の制御にしたがって、入力部１０１からのＪａｖａバイトコードの、Ｊａｖａ仮想マシン８３への供給を制御するようになされている。

【０１４４】以上のように構成されるプログラム実行システムでは、まず最初に、入力部１０１において、図１４の入力部８１における場合と同様にして、署名付きバイトコードおよび暗号化公開キーＡ、並びに公開キーＢが取得される。そして、入力部１０１は、署名付きバイトコードを、署名とＪａｖａバイトコードとに分離し、署名を、署名確認部１０３に、Ｊａｖａバイトコードを、メッセージダイジェストシステム１０２および仮想マシン入力制御部１０４に、それぞれ供給する。さらに、入力部１０１は、暗号化公開キーＡと公開キーＢを復号部８４に供給する。復号部８４では、図１４で説明したように、暗号化公開キーＡが、公開キーＢを用いて、公開キーＡに復号化され、署名確認部１０３に供給される。

【０１４５】一方、メッセージダイジェストシステム１０２では、入力部１０１からのＪａｖａバイトコードからダイジェストが作成され、署名確認部１０３に供給される。署名確認部１０３では、復号部８４からの公開キーＡを用いて、入力部１０１からの署名が復号化されてダイジェストとされる。さらに、署名確認部１０３では、その復号化したダイジェストが、メッセージダイジェストシステム１０２からのダイジェストと比較され、それが一致するかどうかで、入力部１０１からの署名の正当性が確認される。

【０１４６】署名の正当性が確認された場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム１０２からのダイジェストと一致する場合、署名確認部１０３は、入力部１０１からのＪａｖａバイトコードを、Ｊａｖａ仮想マシン８３に出力するように、仮想マシン入力制御部１０４を制御する。この場合、仮想マシン入力制御部１０４は、署名確認部１０３の制御にしたがい、入力部１０１からのＪａｖａバイト

コードを、Ｊａｖａ仮想マシン８３に供給する。
【０１４７】従って、この場合、Ｊａｖａ仮想マシン８３では、入力部１０１から仮想マシン入力制御部１０４を介して供給されるＪａｖａバイトコードが解釈、実行される。
【０１４８】一方、署名の正当性が確認されなかった場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム１０２からのダイジェストと一致しない場合、署名確認部１０３は、入力部１０１からのＪａｖａバイトコードを、Ｊａｖａ仮想マシン８３に出力しないように、仮想マシン入力制御部１０４を制御する。
【０１４９】この場合、仮想マシン入力制御部１０４からＪａｖａ仮想マシン８３に対しては、入力部１０１からのＪａｖａバイトコードは出力されず、従って、Ｊａｖａ仮想マシン８３では、特に処理は行われない。
【０１５０】以上から、アプリケーションプログラムの認証として署名を付す場合も、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードであって、プログラム認証機関で認証されていないものの、Ｊａｖａ仮想マシン８３が実装されたユーザ端末３３を有するユーザへの配布を制限することができる。即ち、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードの、そのＪａｖａ仮想マシン８３が実装されたユーザ端末３３を有するユーザへの配布は、プログラム認証機関とライセンス契約を結んだソフトウェア開発者にのみ許可することができ、Ｊａｖａ仮想マシン８３の開発者は、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードの配布を希望するソフトウェア開発者から、Ｊａｖａ仮想マシン８３を使用するためのライセンス料を得ることが可能となる。
【０１５１】また、署名を付す場合においては、上述したように、その署名を付したＪａｖａバイトコードを改竄したものの、仮想マシン８３上での実行を制限することなどができる。
【０１５２】なお、図１７の実施の形態では、Ｊａｖａ仮想マシン８３に対するＪａｖａバイトコードの入力は、仮想マシン入力制御部１０４からのみ行うことができるようにしておく必要がある。また、署名確認部１０３への公開キーＡの入力は、復号部８４のみから行うことができるようにするのが望ましい。
【０１５３】ここで、Ｊａｖａバイトコードに署名を付す場合においては、Ｊａｖａバイトコードを暗号化する場合と異なり、Ｊａｖａバイトコードそのものが存在する。従って、プログラム実行システムが署名の正当性を確認しないもの（例えば、入力部１０１の出力であるＪａｖａバイトコードが、Ｊａｖａ仮想マシン８３に直接入力されるような構成のもの）であれば、そのプログラム実行システムでは、何の制限もなく、Ｊａｖａバイトコードを解釈して実行することができる。
【０１５４】即ち、逆にいえば、Ｊａｖａバイトコードに署名を付す場合において、その実行を制限したいＪａｖａ仮想マシンの開発者や販売者、さらには、ユーザ端末３３にＪａｖａ仮想マシンを実装して販売する者などは、そのプログラム実行システムを、図１７に示したように構成すれば良いし、その実行の制限を特に希望しない者は、プログラム実行システムを署名の正当性を確認しないような構成とすれば良い。
【０１５５】次に、図５の実施の形態では、プログラム認証機関からユーザに対して、ソフトウェアであるプログラム実行システム（ユーザ端末３３をプログラム実行システムとして機能させるためのプログラム）を、そのままユーザ端末３３に実装することができる形で提供するようにしたが、このプログラム実行システムも、ソフトウェア開発者が提供するアプリケーションプログラムと同様に、暗号化したり、また署名を付して提供するようにすることができる。
【０１５６】例えば、プログラム認証機関からユーザに対して、プログラム実行システムに署名を付加して提供する場合においては、プログラム認証機関サーバ３２においては、例えば、図１８のフローチャートにしたがった処理が行われる。
【０１５７】即ち、この場合、プログラム認証機関サーバ３２では、まず最初に、ステップＳ３１において、プログラム実行システムとしてのプログラムが、ユーザ端末３３のＣＰＵ６１が実行可能なコード（以下、適宜、実行コードという）にコンパイルされる。このコンパイルは、ＣＰＵ５１によって行われる。そして、ＣＰＵ５１は、ステップＳ３２において、ステップＳ３１のコンパイルの結果得られた実行コードから、例えば、図１０で説明したようにしてダイジェストを作成し、ステップＳ３３に進む。ステップＳ３３では、ＣＰＵ５１において、ステップＳ３２で作成されたダイジェストが、例えば、公開キーＡの暗号化に用いられる秘密キーＢを用いて暗号化され、これにより署名（ディジタルサイン）が作成される。そして、ステップＳ３４に進み、そのディジタルサインが、実行コードに付され（このようにディジタルサイン（署名）が付された実行コードを、以下、適宜、署名付き実行コードという）、その署名付きバイトコードが、署名の作成に用いられた秘密キーＢと対になる公開キーＢと対応付けられて、補助記憶装置５６に記憶され、処理を終了する。
【０１５８】次に、図１９は、ユーザ端末３３において、上述したような署名が付加されたプログラム実行システム（署名付き実行コード）を実装するためのローダ（プログラム実行システムの実装装置）の機能的構成例を示している。
【０１５９】この実施の形態では、ローダは、入力部１１１、メッセージダイジェストシステム１１２、署名確認部１１３、および実行システム１１４で構成され、入

力部１１１、メッセージダイジェストシステム１１２、または署名確認部１１３は、図１７の入力部１０１、メッセージダイジェストシステム１０２、または署名確認部１０３とそれぞれ同様に構成されている。実行システム１１４は、ユーザ端末３３のＣＰＵ６１その他の実行コードを解釈して実行する部分に相当する。

【０１６０】以上のように構成されるローダでは、入力部１１１において、署名付き実行コードおよび公開キーＢが取得される。そして、入力部１１１は、署名付き実行コードを、署名と実行コードとに分離し、署名を署名確認部１１３に、実行コードをメッセージダイジェストシステム１１２および実行システム１１４に供給する。さらに、入力部１１１は、公開キーＢを署名確認部１１３に供給する。

【０１６１】メッセージダイジェストシステム１１２では、入力部１１１からの実行コードからダイジェストが作成され、署名確認部１１３に供給される。署名確認部１１３では、入力部１１１からの公開キーＢを用いて、入力部１１１からの署名が復号化されてダイジェストとされる。さらに、署名確認部１１３では、その復号化したダイジェストが、メッセージダイジェストシステム１１２からのダイジェストと比較され、それが一致するかどうかで、入力部１１１からの署名の正当性が確認される。

【０１６２】署名の正当性が確認された場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム１１２からのダイジェストと一致する場合、署名確認部１１３は、入力部１１１からの実行コードを解釈して実行するように、実行システム１１４を制御する。この場合、実行システム１１４では、署名確認部１１３の制御にしたがい、入力部１１１からの実行コードが解釈、実行される。

【０１６３】一方、署名の正当性が確認されなかった場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム１１２からのダイジェストと一致しない場合、署名確認部１１３は、入力部１１１からの実行コードを無視するように、実行システム１１４を制御する。この場合、実行システム１１４では、署名確認部１１３の制御にしたがい、入力部１１１からの実行コードが無視され、従って、特に処理は行われない。

【０１６４】以上のように、プログラム実行システムに署名を付加する場合においては、その改竄などを防止することが可能となる。

【０１６５】なお、上述したように、プログラム実行システムは暗号化することも可能であり、この場合、プログラム実行システムのリバースエンジニアリングを防止することが可能となる。

【０１６６】プログラム実行システムに付加された署名の正当性が確認された場合、上述したように、実行システム１１４において入力部１１１からの実行コードが解釈、実行され、これにより、ユーザ端末３３には、例えば、図２０に示すように、図１７における場合と同様のプログラム実行システムが実装される。但し、この場合においては、復号部８４への公開キーＡの供給は、入力部１０１から行われるのではなく、図１９のローダにおける入力部１１１から行われる。

【０１６７】ところで、アプリケーションプログラムや、プログラム実行システムのプログラムなどのように、比較的情報量の多いものを、例えば、ＲＳＡ方式などの公開鍵暗号化方式で暗号化し、また、その復号化を行う場合においては、その処理に時間を要する。これに対して、ＤＥＳ方式などの共通鍵暗号化方式による暗号化／復号化は、情報量が多いものでも、比較的短時間に行うことができる。一方、ＲＳＡ方式では、暗号化に用いる秘密キーと、その復号化に用いる公開キーとが異なるので、公開キーは、上述したように一般に公開しても問題はないが、ＤＥＳ方式では、暗号化と復号化とに同一の共通キーを用いるため、共通キーは、当事者以外の者に知られないように、厳重に管理する必要がある。

【０１６８】そこで、暗号化／復号化の処理を短時間で行うとともに、キーの管理を容易にする手法として、例えば、アプリケーションなどの比較的情報量の多いものの暗号化には、ＤＥＳ方式を採用し、そのＤＥＳ方式の暗号化に用いる共通キーを、ＲＳＡ方式で暗号化するものがある（このようにＤＥＳ方式とＲＳＡ方式とを組み合わせた暗号化の手法を、以下、適宜、組合せ方式という）。

【０１６９】即ち、図２１は、アプリケーションプログラムについて、組合せ方式による暗号化を行う場合のソフトウェア開発者サーバ３１の機能的構成例を示している。

【０１７０】アプリケーションプログラムをＪａｖａコンパイラでコンパイルして得られるＪａｖａバイトコードは、暗号化器１２１に入力される。暗号化器１２１には、Ｊａｖａバイトコードの他に、共通キーも入力されるようになされており、そこでは、Ｊａｖａバイトコードが、共通キーを用いて、例えば、ＤＥＳ方式により暗号化され、これにより、暗号化バイトコード（暗号文）とされる。

【０１７１】暗号化器１２１に入力される共通キーは、暗号化器１２２にも入力されるようになされている。暗号化器１２１は、例えば、ＲＳＡ方式による暗号化を行うもので、そこでは、共通キーが、秘密キーＡを用いて暗号化される。

【０１７２】この場合、ソフトウェア開発者からユーザに対しては、暗号化バイトコード、秘密キーＡで暗号化された共通キー（以下、適宜、暗号化共通キーという）、およびプログラム認証機関から取得した暗号化公開キーＡが１セットとして配布される。

【０１７３】次に、図２２は、アプリケーションプログ

ラムの暗号化が組合せ方式で行われる場合のユーザ端末３３に実装されるプログラム実行システムの機能的構成例を示している。なお、図中、図１４における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【０１７４】復号部１３１は、例えばＲＳＡ方式により、暗号化共通キーを復号化するようになされており、その復号化結果としての共通キーを、復号部１３２に供給するようになされている。復号部１３２は、例えばＤＥＳ方式により、暗号化バイトコードを復号化するようになされている。

【０１７５】以上のように構成されるプログラム実行システムでは、入力部８１において、暗号化バイトコード（暗号化されたＪａｖａバイトコード）、暗号化共通キー（暗号化された共通キー）、および暗号化公開キーＡ（暗号化された公開キーＡ）、並びに公開キーＢが取得される。そして、入力部８１は、暗号化公開キーＡおよび公開キーＢを復号部８４に、暗号化共通キーを復号部１３１に、暗号化バイトコードを復号部１３２に、それぞれ出力する。

【０１７６】復号部８４では、上述したように、暗号化公開キーＡが、公開キーＢを用いて、公開キーＡに復号化される。そして、この公開キーＡは、復号部１３１に供給される。復号部１３１では、入力部８１からの暗号化共通キーが、復号部８４からの公開キーＡを用いて、共通キーに復号化され、復号部１３２に供給される。復号部１３２では、入力部８１からの暗号化バイトコードが、復号部１３１からの共通キーを用いて、Ｊａｖａバイトコードに復号化され、Ｊａｖａ仮想マシン８３に供給される。

【０１７７】以上のような組合せ方式によれば、キーの管理の簡単化および暗号化／復号化処理の高速化を図ることが可能となる。

【０１７８】また、組合せ方式は、アプリケーションプログラムを暗号化する場合の他、例えば、プログラム実行システムのプログラム（実行コード）を暗号化する場合などにも適用可能である。

【０１７９】なお、本発明は、上述したインタープリタ形式およびＪＩＴコンパイラ形式のいずれのＪａｖａ仮想マシンにも適用可能であるし、また、Ｊａｖａ仮想マシン以外の仮想マシン、さらには、例えば、Ｃ言語や、Ｃ＋＋言語などの処理系のようにプログラム実行システムへの入力が機械語コードの場合や、Ｂａｓｉｃ言語の処理系のようにプログラム実行システムへの入力がソースコードの場合などについても適用可能である。

【０１８０】また、図１４や図１７などの実施の形態においては、プログラム実行システムを１つしか設けていないが、ユーザ端末３３には、このプログラム実行システムを複数設けることも可能である。例えば、入力部８１や１０１を複数設けた場合においては、複数経路からの暗号文や署名付きバイトコードの入力が可能となる。さらに、例えば、復号部８２や８４を複数設けた場合においては、複数の復号アルゴリズムにしたがって暗号文を復号することが可能となる。また、例えば、Ｊａｖａ仮想マシン８３を複数設けた場合には、複数のＪａｖａバイトコード形式をサポートすることが可能となる。さらに、例えば、メッセージダイジェストシステム１０２および署名確認部１０３を複数設けた場合には、複数の手法それぞれで付された複数の署名の確認を行うことが可能となる。

【０１８１】また、本実施の形態では、１のキーを用いて暗号化や署名の作成を行うようにしたが、暗号化や署名の作成は、複数のキーを用いて行うことも可能である。即ち、例えば、暗号化は、複数のキーを順次用いて行うことができ、また、署名は、複数のキーに対応する数だけ作成するようにすることなどができる。

【０１８２】なお、本実施の形態におけるＪａｖａバイトコードとは、例えば、Java Application, Java Applet, Java Benas, Java Class Libraryその他の多数の形態のＪａｖａバイトコードをすべて含むものである。

【０１８３】

【発明の効果】請求項１に記載の情報処理装置および請求項５に記載の情報処理方法によれば、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものが、第２のキーを用いて復号化され、その復号化された第１のキーを用いて、プログラムを暗号化したものが復号化される。そして、その復号化されたプログラムが実行される。また、請求項６に記載の記録媒体には、コンピュータに、プログラムを暗号化したものを復号化するのに必要な第１のキーを暗号化したものを、第２のキーを用いて復号化させ、その復号化された第１のキーを用いて、プログラムを暗号化したものを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されている。従って、第１のキーが復号化され、さらに、プログラムが復号化された場合のみ、そのプログラムが実行されるようにすることが可能となる。

【０１８４】請求項８に記載の情報処理装置および請求項９に記載の情報処理方法によれば、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムが暗号化される。また、請求項１０に記載の記録媒体には、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されている。従って、請求項１に記載の情報処理装置で実行可能な、暗号化されたプログラムの提供が可能となる。

【０１８５】請求項１１に記載の情報処理装置および請求項１２に記載の情報処理方法によれば、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものが、第２のキーを用いて復号化され、そ

の復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかが確認される。そして、署名が正当なものであることが確認された場合のみ、プログラムが実行される。また、請求項１３に記載の記録媒体には、コンピュータに、プログラムに付されている署名を確認するときに用いる第１のキーを暗号化したものを、第２のキーを用いて復号化させ、その復号化された第１のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認させ、署名が正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されている。従って、第１のキーが復号化され、さらに、プログラムに付されている署名が正当なものである場合のみ、そのプログラムが実行されるようにすることが可能となる。

【０１８６】請求項１４に記載の情報処理装置および請求項１５に記載の情報処理方法によれば、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムが処理される。また、請求項１６に記載の記録媒体には、請求項１１に記載の情報処理装置において署名が正当なものであると確認されるように処理されたプログラムが記録されている。従って、請求項１１に記載の情報処理装置で実行可能なように処理されたプログラムの提供が可能となる。

【図面の簡単な説明】

【図１】計算機１の資源と、そこに実装されたＪａｖａ仮想マシン１１の資源との対応関係を示す図である。

【図２】Ｊａｖａ仮想マシン１１の処理を説明するための図である。

【図３】Ｊａｖａ仮想マシン１１の処理を説明するための図である。

【図４】Ｊａｖａ仮想マシン１１の処理を説明するための図である。

【図５】本発明を適用したプログラム提供システムの一実施の形態の構成例を示すブロック図である。

【図６】図５のソフトウェア開発者サーバ３１の構成例を示すブロック図である。

【図７】図５のプログラム認証機関サーバ３２の構成例を示すブロック図である。

【図８】図５のユーザ端末３３の構成例を示すブロック図である。

【図９】暗号化／復号化システムの構成例を示すブロック図である。

【図１０】ディジタルサインを用いた暗号化／復号化システムの構成例を示すブロック図である。

【図１１】ソフトウェア開発者サーバ３１の処理を説明するためのフローチャートである。

【図１２】ソフトウェア開発者サーバ３１の処理を説明するためのフローチャートである。

【図１３】プログラム認証機関サーバ３２の処理を説明するためのフローチャートである。

【図１４】プログラム実行システムの第１の機能的構成例を示すブロック図である。

【図１５】プログラム実行システムの第２の機能的構成例を示すブロック図である。

【図１６】ソフトウェア開発者サーバ３１の処理を説明するためのフローチャートである。

【図１７】プログラム実行システムの第３の機能的構成例を示すブロック図である。

【図１８】プログラム認証機関サーバ３２の処理を説明するためのフローチャートである。

【図１９】ローダの機能的構成例を示すブロック図である。

【図２０】プログラム実行システムの第４の機能的構成例を示すブロック図である。

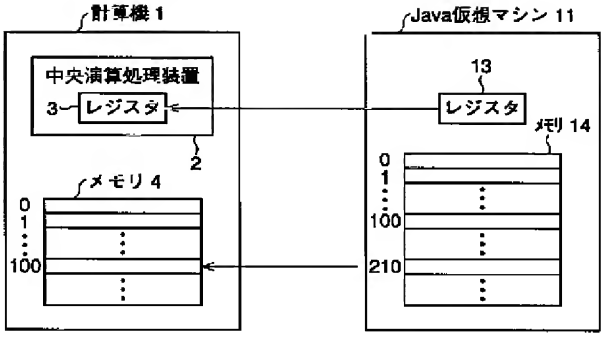
【図２１】ソフトウェア開発者サーバ３１の機能的構成例を示すブロック図である。

【図２２】プログラム実行システムの第５の機能的構成例を示すブロック図である。

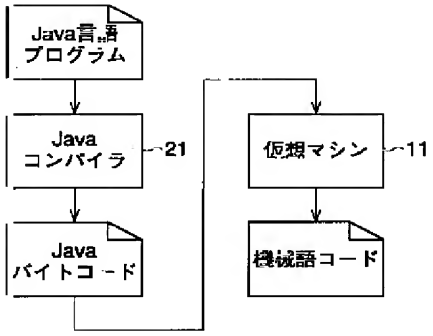
【符号の説明】

１　計算機，　２　中央演算処理装置，　３　レジスタ，　４　メモリ，　１１　Ｊａｖａ仮想マシン，　１３　レジスタ，　１４　メモリ，　２１　Ｊａｖａコンパイラ，　３１　ソフトウェア開発者サーバ，　３２　プログラム認証機関サーバ，　３３　ユーザ端末，　３４　ネットワーク，　３５　記録媒体，４１　ＣＰＵ，　４２　ＲＯＭ，　４３　ＲＡＭ，　４４　入力部，　４５　出力部，　４６　補助記憶装置，　４７　通信制御部，　５１　ＣＰＵ，　５２ＲＯＭ，　５３　ＲＡＭ，　５４　入力部，　５５　出力部，　５６　補助記憶装置，　５７　通信制御部，　６１　ＣＰＵ，　６２　ＲＯＭ，　６３　ＲＡＭ，　６４　入力部，　６５　出力部，　６６　補助記憶装置，　６７　通信制御部，　７１　暗号化器，　７２　復号化器，　８１　入力部，　８２　復号部，８３　Ｊａｖａ仮想マシン，　８４　復号部，　９１　ダイジェスト作成器，９２　暗号化器，　９３　復号化器，　９４　ダイジェスト作成器，　９５署名確認器，　１０１　入力部，　１０２　メッセージダイジェストシステム，１０３　署名確認部，　１０４　仮想マシン入力制御部，　１１１　入力部，１１２　メッセージダイジェストシステム，　１１３　署名確認部，　１１４実行システム，　１２１，１２２　暗号化器，　１３１，１３２　復号部

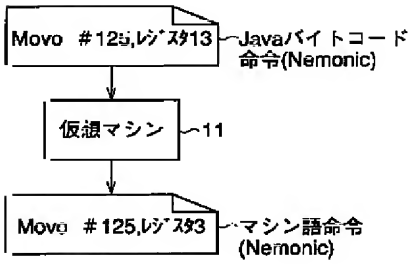
【図１】



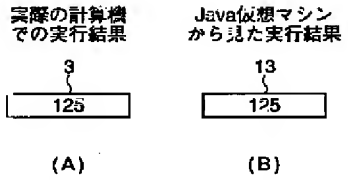
【図２】



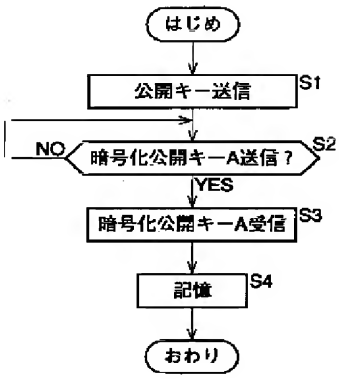
【図３】



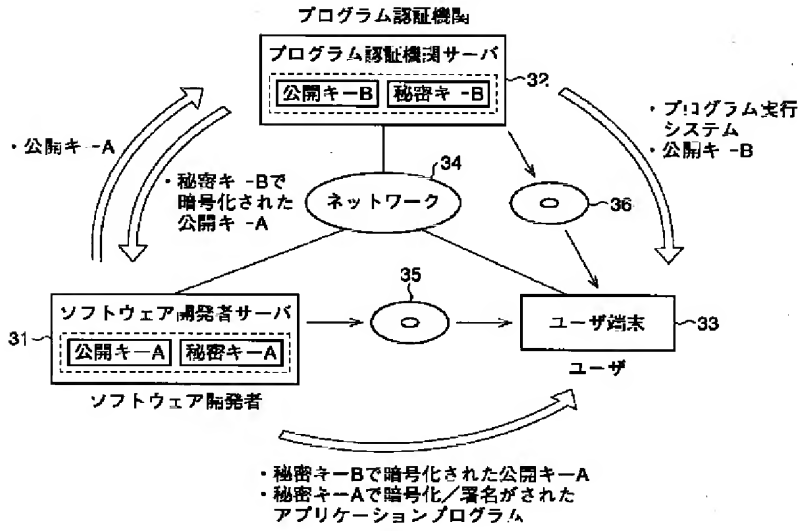
【図４】



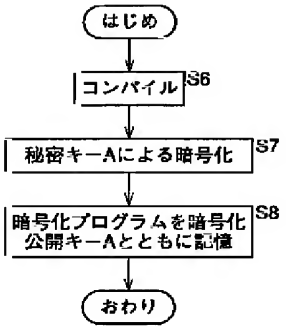
【図１１】



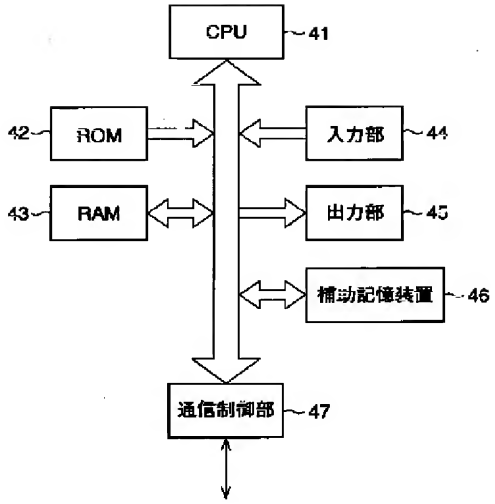
【図５】



【図１２】

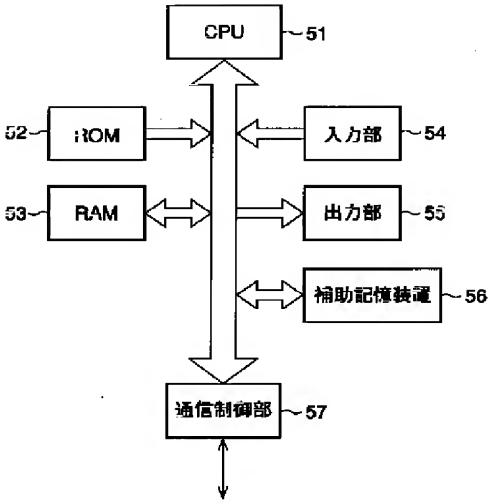


【図６】



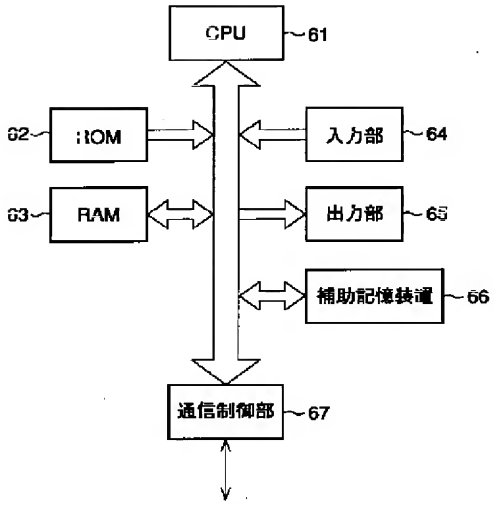
ソフトウェア開発者サーバ 31

【図７】



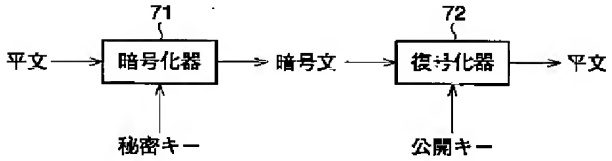
プログラム認証機関サーバ 32

【図８】



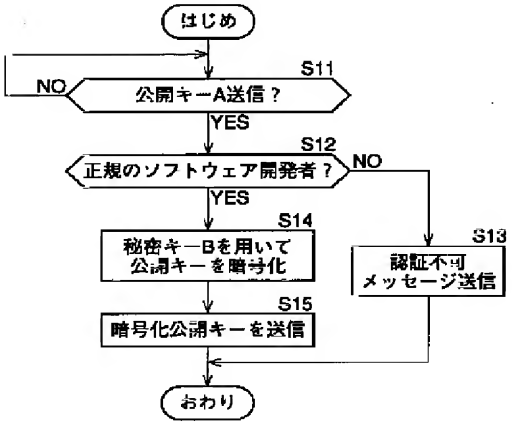
ユーザ端末 33

【図９】

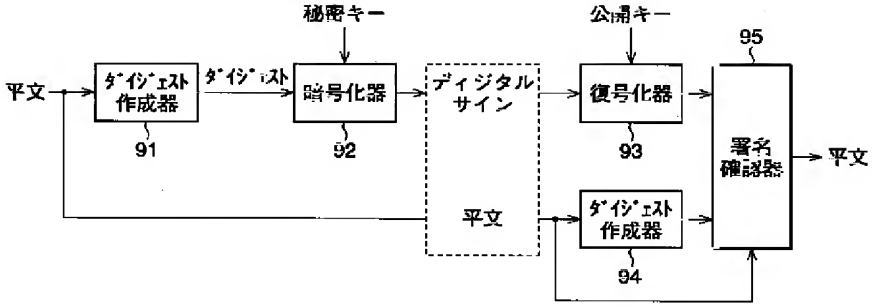


公開キー暗号化／復号化システム

【図１３】

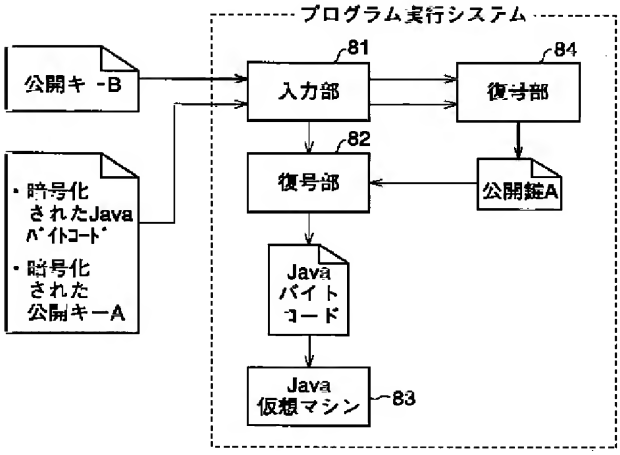


【図１０】

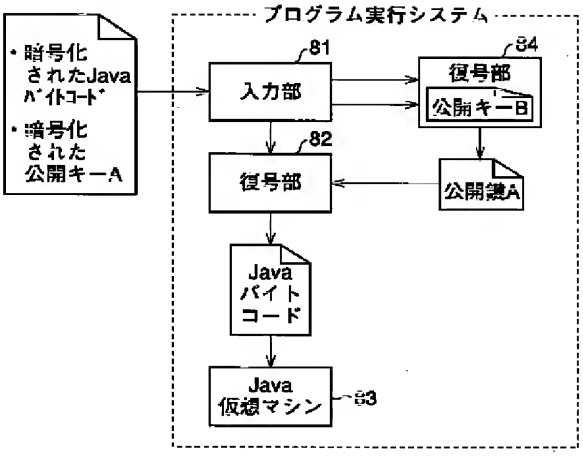


ディジタルサインを用いた暗号化／復号化システム

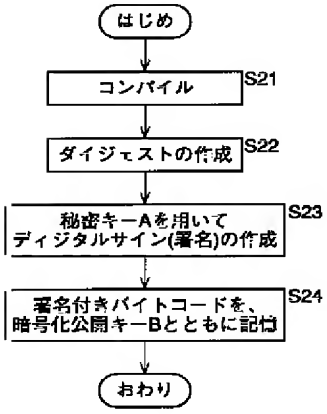
【図１４】



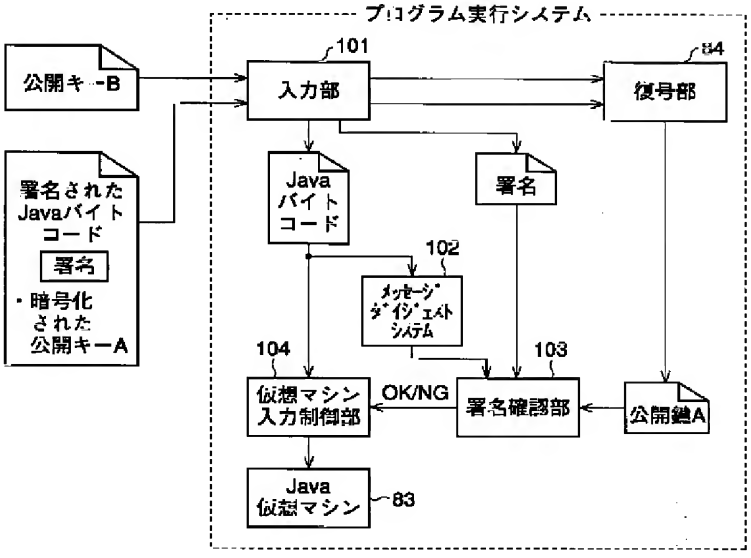
【図１５】



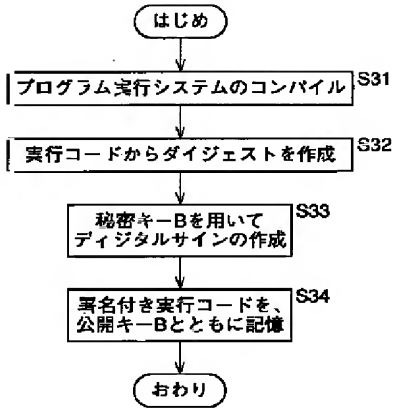
【図１６】



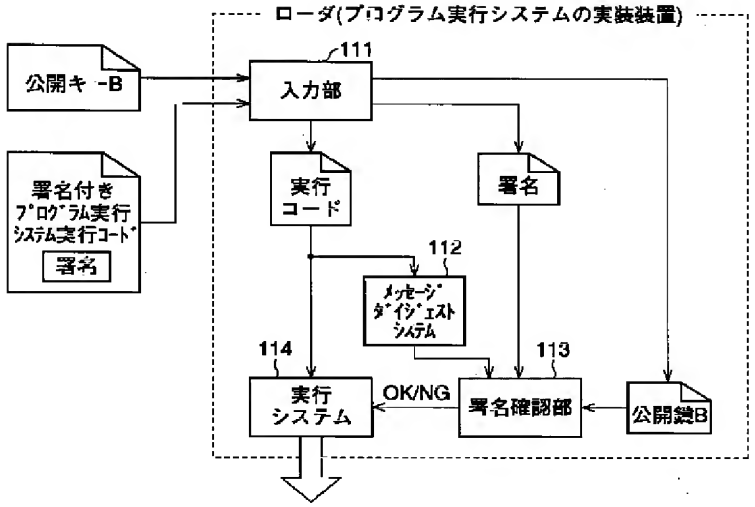
【図１７】



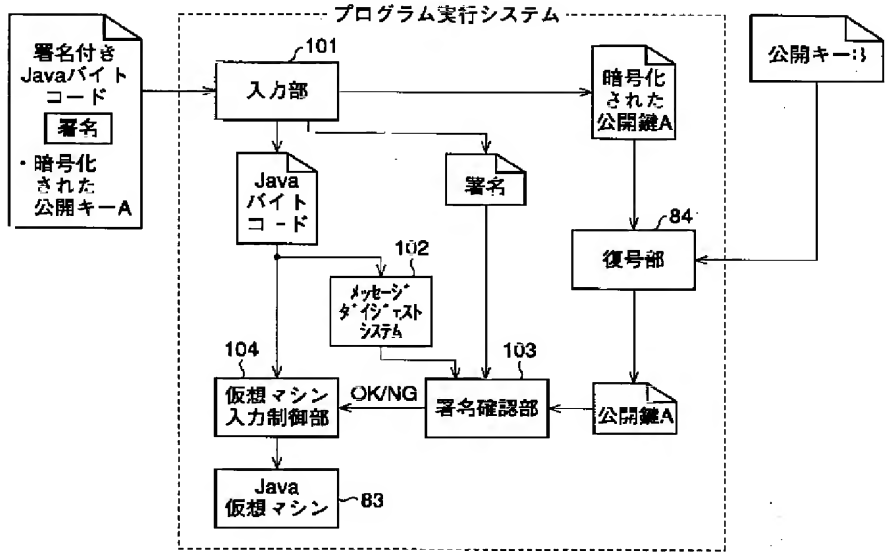
【図１８】



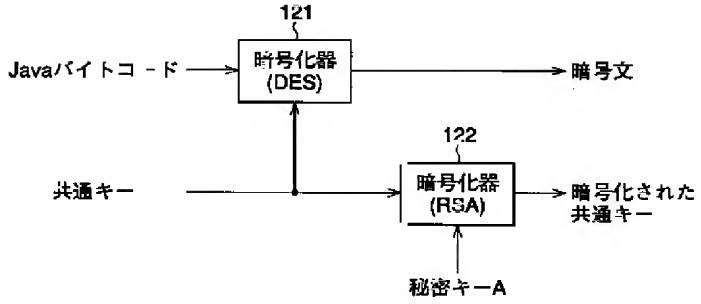
【図１９】



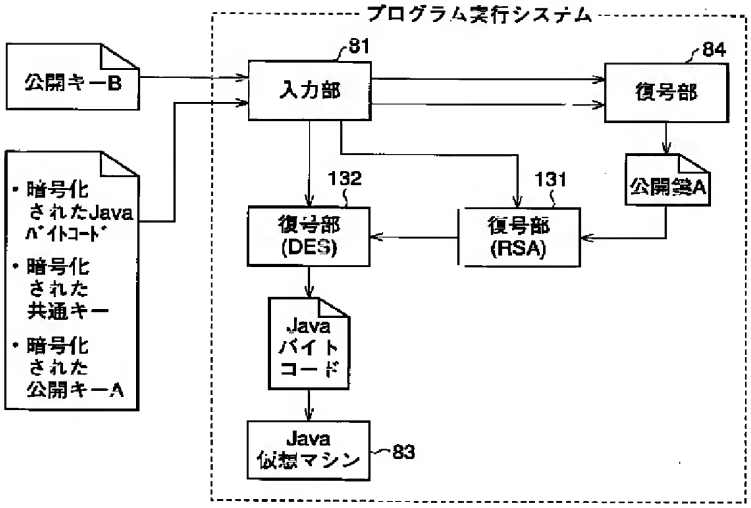
【図２０】



【図２１】



【図２２】


プログラム実行システム
81
入力部
84
復号部
公開キーB
・暗号化されたJavaバイトコード
・暗号化された共通キー
・暗号化された公開キーA
132
131
公開鍵A
復号部(DES)
復号部(RSA)
Javaバイトコード
Java仮想マシン
83